

# The NIST SAMATE Project

Paul E. Black

National Institute of Standards and Technology

<http://samate.nist.gov/>

paul.black@nist.gov



# What is the Project?



- **Software Assurance Metrics And Tool Evaluation (SAMATE) project is sponsored in part by DHS**
- **Research metrics to assess the effectiveness of tools and techniques for software assurance**
- **Leads to improved tools and provides assurance to users of the benefits of the tools**
- **Web site     <http://samate.nist.gov/>**



# Current Areas of Concentration

- **Web application scanners**
- **Source code security analyzers**
- **Work with MITRE and others to define “ground truth” for software assurance**
- **Tool effectiveness studies**
  - Including SATE
- **Software labels**

# Moving Forward

- **Develop additional test suites for assessing web application scanners**
- **Investigate deeper problems in source code analysis during annual Static Analysis Tool Exposition (SATE)**
- **More precisely describe weaknesses, etc.**
- **Improve utility of the SAMATE Reference Dataset (SRD) repository through:**
  - **Providing guidance in running tools against test cases**
  - **Establishing quality levels of many test cases**
  - **Broadening content in areas of language, size, and application**

# Further Out

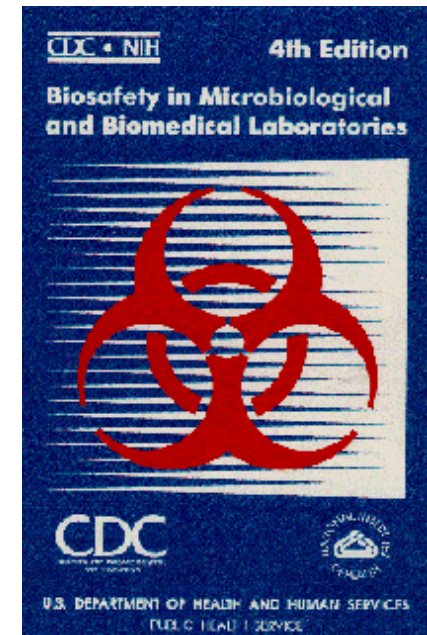
- **Binary analyzers**
- **Higher level (design & architecture) tools**
- **Statistical analysis and mathematical modeling, e.g., bug rediscovery and capture/recapture with non-uniform distributions**

# Static Analysis Tool Exposition (SATE) Overview

- Advance research in, and improvement of, static analysis tools for security-relevant defects and speed tool adoption by demonstrating their use on real software.
- Briefly, participants run their tools on chosen programs. NIST-led researchers analyze tool reports. Results and experiences are shared at a workshop. The tool reports and analysis are made publicly available later.
- **2008 Participants:**
  - Aspect Security ASC
  - Checkmarx CxSuite
  - Flawfinder
  - Fortify SCA
  - Grammatech CodeSonar
  - HP DevInspect
  - SofCheck Inspector for Java
  - UMD FindBugs
  - Veracode SecurityReview
- **2009 Participants:**
  - Armorize
  - Checkmarx CxSuite
  - Coverity
  - Grammatech CodeSonar
  - Klocwork
  - LDRA
  - SofCheck Inspector for Java
  - Veracode SecurityReview
- **2010: Organizing meeting this morning**

# Researching Risky Software

- Many people research malware, but there are no widely accepted protocols.
- Biological research has defined levels with associated practices, safety equipment, and facilities.
- Some approaches are
  - Weakened programs (auxotrophs)
  - Programs that **ALERT**
  - Outgoing firewalls
  - Isolated networks



# Software Reference Dataset

The screenshot shows the SAMATE NIST Software Reference Dataset search interface. The header includes the SAMATE logo, NIST logo, and DHS National Cyber Security Division logo. The navigation bar contains links: SRD Home, View / Download, Search / Download, More Downloads, Submit, and Test Suites. The search interface has two tabs: Extended Search and Source Code Search. The search form includes fields for Number (Test case ID), Description contains, Contributor/Author, Bad / Good (Any...), Language (Any...), Type of Artifact (Any...), Status (Candidate, Approved), Weakness (Any...), Code complexity (Any...), and Date (Any, Before, After). A Search Test Cases button is at the bottom. The results pane on the right shows a list of weaknesses and code complexity categories, including CWE-485: Insufficient Encapsulation, CWE-388: Error Handling, CWE-389: Error Conditions, Return Values, Status Codes, CWE-254: Security Features, CWE-227: Failure to Fulfill API Contract (API Abuse), CWE-019: Data Handling, CWE-361: Time and State, CWE-398: Indicator of Poor Code Quality, CWE-470: Use of Externally-Controlled Input to Select Classes, CWE-465: Pointer Issues, CWE-411: Resource Locking Problems, CWE-401: Failure to Release Memory Before Removing Last, CWE-415: Double Free, CWE-416: Use After Free, and CWE-417: Channel and Path Errors.

- Public repository for software test cases
- Almost 1800 cases in C, C++, Java, and Python
- Search and compose custom Test Suites
- Contributions from Fortify, Defence R&D Canada, Klocwork, MIT Lincoln Laboratory, Praxis, Secure Software, etc.